Design of Word Input Prediction System Based on LSTM

Yuerong Ma, Yating Wang, Chengjie Wan, Tian Qiao
College of Communication and Art Design, University of Shanghai for Science and Technology, Shanghai, China

Email address:
2431450041@qq.com(Yuerong Ma), 1612807336@qq.com (Yating Wang), 2845365434@qq.com (Chengjie Wan)
1486503287@qq.com (Tian Qiao)

Abstract: At present, when the input method software is applied in different industries, the correlation between the predicted next word and the user input word is not high and the accuracy is low. This paper proposes a word input prediction system based on LSTM that can be applied in specific fields to improve the correlation of predicted words. The long and short-term memory network (LSTM) is used to train text data sets in different fields to generate specialized models, which can be used to design word input prediction systems in specific fields.

Keywords: Short and long-term memory Network (LSTM); Word input prediction; System design

1. Introduction

The advent of the new media era has not only changed people's way of life, but also greatly impacted and changed all walks of life. In order to enhance the competitiveness of the industry, all industries are exploring new models, and mobile terminals and mobile applications are widely promoted and applied. For example, in the field of news, you can follow real-time reports of various kinds of news through weibo app, and in the field of medical, you can communicate with doctors online through medical app. In order to attract and retain more audiences, the efficiency of journalists' writing and the speed of doctors' online responses are particularly important. Word input prediction can predict the next word according to the context of the input word, and the system will prompt a list of related word suggestions for the user to choose, thus reducing the input time consumption. Therefore, word input prediction can be applied to various input systems to improve input efficiency.

The traditional N-Gram model is based on the Markov chain model to measure the probability of the occurrence of the next word in the n-1 word sequence [1, 2], but it will lead to data sparseness and dimension disaster and fail to reflect the semantic differences between words. Naive Bayes method [3] outputs the terms with the maximum posteriori probability based on the joint probability distribution of the terms with independent hypothesis of feature conditions by using Bayes' theorem. Its deficiency lies in the poor accuracy of prediction. Latent Semantic Analysis (LSA)[4] makes word prediction through lexical relational Semantic Analysis of text, but its inference is only based on text frequency, regardless of word order and text grammar.With the development and implementation of deep learning technologies[5], Convolutional Neural Networks (CNN), Recurrent Neural Network (RNN), and Long short-term Memory Network (LSTM) are proposed. LSTM [6] and other nonlinear sequence models can effectively deal with serialized data to study prediction problems. LSTM[7,8,9], as a typical time series prediction model, takes long-term memory and short-term memory into account in its memory mechanism, and can make use of long-distance temporal information to predict post-sequential time series. Therefore, the predicted words are in line with the specific user's vocabulary habits.

Therefore, this paper designs a word input prediction system based on LSTM. Firstly, the text data set of a specific industry is preprocessed, and LSTM network is used to train the preprocessed text to generate the corresponding industry vocabulary prediction model, which is applied to the input system of the industry.

2. Related Work

In recent years, a large number of researchers have carried out research on word input prediction. Typical word prediction methods include N-Gram model, RNN model, LSTM model and so on.

Word prediction based on statistical language model makes use of the probability distribution of words in a sequence. Yazdani A [10] et al. introduced the Tri-Gram language model for text prediction and reduced the typing time consumption of users when they input text by displaying the suggested list of the next word. Henrique X[11] et al. proposed a hybrid word prediction model based on naive Bayes and latent semantic analysis (LSA) theory, which took syntactic/semantic rules between words into consideration to reduce the training time, and optimized parameters by gradient descent technology to improve the accuracy of prediction. Hasan [12]et al. proposed a new term "mutual context", which generates mutual context through the four context attributes of both users, considers the relevance between users, and predicts the next word in a more accurate order.

The method based on deep learning uses corpus to train neural network to predict words. Mikolov [13] et al. proposed a recurrent neural network language model for word prediction, which can accurately predict the next word through word embedding of the input current word and superposition of the previous state. Zhou[14] et al. proposed a mixed word prediction model, C-LSTM, to predict the next word more accurately by capturing the local features of phrases and the global features and temporal features of sentences. Wang[15] et al. proposed a new convolution structure, genCNN, which integrates local correlation and global correlation in word sequences to predict the next word with variable length word history. Zhao L C [16] et al. LSTM network based on BERT feature was introduced for sentence recommendation to improve the text input efficiency and quality of electronic medical records. Sukhbataa[17] et al. introduced recursive neural network to predict the next word in the text sequence and reduced the supervision needed in the training process through end-to-end training, which is more generally applicable to the realistic environment of word input prediction. Shalini Ghosh et al. [18] proposed CLSTM (Context LSTM) model to improve the performance of word input prediction by attaching context vectors to the input terms and using terms and topics as features.

3. System design flow framework

The flow framework of LSTM-based word input prediction system is shown in Figure 1, which is mainly divided into offline phase and online phase. The offline phase is mainly responsible for preprocessing the text of the industry dataset, while the online phase is mainly to get the most relevant word sequence according to the words input by users in a specific industry.

3. 1 Offline phase

The specific steps of text preprocessing in offline phase are as follows:

Step 1: The dataset text R is segmented according to the separated sentence end character, and R_1=[S_1, S_2, S_3,…, S_n] is obtained;

Step 2: The text R_1 is cleaned to stop words, numbers and special characters in turn, andR_2=[S_1, S_2, S_3,…, S_n] is obtained;

Step 3: Porter Stemmer algorithm is used for word segmentation of sentences separated by R_2, namely S_i=[w_1,w_2,w_3,…,w_m], and R_3=[w_1,w_2,w_3,…,w_n] is obtained;

3.2 Online phase

Specific steps as shown in Figure 2, firstly we use text data after preprocessing to train LSTM model until finished andthe model is generated, Based on the words entered by the user, the model predicts the next word and sorts it from most likely to least likely. The model takes the top*k*word suggestion list and recommends it to the user. The user can select appropriate words from the word suggestion list. If the input is not finished, the model predicts related words again according to the existing input sequence until the user completes the input task.

4. Word input prediction system

The key of system design is to construct a suitable word input prediction model. In the process of user input, the input words have the order before and after, and the input sequence of LSTM also has the recursive structure in the time dimension. Therefore, LSTM is very suitable for word input prediction.

4.1 Long and short term memory network

The proposal of LSTM effectively solves the problems of gradient disappearance and gradient explosion of cyclic neural network in the process of long sequence training. LSTM can store the previous state characteristics through the gated control structure, so it has a special memory mechanism, can effectively deal with the problem of multiple variables, and can learn long-term dependent information, and has excellent performance in long sequences.

The structure of LSTM is shown in Figure 3, which is composed of input gate, forget gate, output gate and cell state.The key is that the cell state extends directly to the whole chain, and some small linear interactions are used to facilitate the flow of information along the original. LSTM protects and controls cell state by adding or deleting information through a gate composed of sigmoid layer and point-by-point multiplication [19].

The first step of LSTM is to determine what information to discard from the cell state, which is determined by the sigmoid layer of the forget gate. The $h_{t-1}$of the above step and that $x_t$of this step are used as inputs, and a number between 0 and 1 is output for each number in the cell stateC_{t-1}. 1 indicates complete retention and 0 indicates complete abandonment.As shown in Equation 1, here $h_{t-1}$is the output of the previous LSTM module, $x_t$is the input of the current LSTM module, $C_{t-1}$is the cell state output of the previous LSTM module, W_f is the weight vector, $b_f$is the bias of the forget gate and f_t is the output of the forget gate.

The next step is to determine what new information to store in the cell state. This consists of two parts: the sigmoid layer, called the input gate layer, determines which values we will update, and the tanh layer creates a vector of new candidate values $\widetilde{C_t}$that can be added to the cell state. The two are then combined to renew the cell state. As shown in Equation 2and3，here$i_t$is the output of the input layer gate,W_i is the weight vector, $b_i$is the bias of the input gatelayer. W_C and $b_C$are the weight vector and bias for creating the new candidate value vector$\widetilde{C_t}$respectively. $C_t$is the updated state of the cell.Asshownin Equation 4, wemultiply $f_t$by the old cell stateC_{t-1}, forgetting what we decided to forget before. We then add $i_t*\widetilde{C_t}$ to scale to get the new candidate based on how big we decide to update each state value.

Finally we need to decide what to output. This output is based on the cell state, but this is the filtered version. First we decide which parts of the cell state to output through a sigmoid layer. We then multiply the cell state through the tanh layer (with values between -1 and 1) by the output of the sigmoid layer to output only the identified portion. As shown in Equation 5 and 6, here $o_t$is the output of the output layer gate, $W_o$and $b_o$are the weight vector and bias of the output gate layer respectively. $h_t$is the output of the current LSTM module.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (1)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (3)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (4)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = o_t * \tanh(C_t) \quad (6)$$

4.2 Construction of word input prediction model

The LSTM-based word input prediction model constructed in this paper is shown in Figure 4, which is mainly composed of two parts. The first part is the network architecture design based on LSTM, which consists of an input layer, a hidden layer, and an output layer. The hidden layer consists of two LSTM layers and a DROPOUT layer. The LSTM network is trained with the preprocessed industry data set text. Build a dictionary by numbering each word according to its frequency. In order to decode the output of LSTM, a reverse dictionary is also generated.The input is the word vector, and the output is the prediction probability vector after Softmax activation function normalization. The loss function is reduced by Adam optimizer[20], and the optimal model is obtained by constantly updating the iterative weight.

The second part is for the user to input words in the system. The model outputs the corresponding prediction probability vector according to the input words. After the probability is sorted in reverse order, the corresponding top k word suggestion list is decoded by the reverse dictionary for the user to choose.

5. Results

The experimental machine is configured with Intel(R) Core(TM) i5-8250Uand 8 GB RAM, under Windows 2016R, and development environment is PyCharm 2019. The datasets used in the experiment are the sports dataset and the financial dataset from BBC Dataset. In the experiment, the LSTM hidden layer node number is set to 512, the initial learning rate is set to 0.001, the data batch processing capacity is set to 128, and the dropout layer loss rate is set to 0.2.

Three words were randomly selected to demonstrate the experimental results more clearly and intuitively. The results of sports news-oriented word input prediction are shown in Table 1. Because the training data set is sports news reports, the prediction results of word input are closely related to sports, and the word sequences returned by the system are more in line with the requirements of sports news writers.For example, when the input “world” vocabulary in turn to “record”, “champion”, “cup”, etc., in the sports news “world record”, “world champion” and “world cup” and so on are common fixed collocation, of correlation exists between the two words is higher, therefore very accord with sports journalists writing habits and requirements. Similarly, “football” and “final” return results are all acceptable to sports journalists.

The results of word input prediction for financial news are shown in Table 2. When the word “financial” is entered, the words returned are “crisis”, “market”, “capital” and so on, and “financial crisis”, “financial market”, “financial capital”are also common phrases in financial news, so it fully conforms to the writing habits and requirements of financial journalists. Therefore, through case study and analysis, the method in this paper can more accurately return the results expected by journalists, thus improving the writing speed of the author.

Therefore, the word input prediction system based on LSTM designed in this paper can be effectively applied to specific fields and improve user input efficiency. By preprocessing the text data set of a specific industry and using LSTM to train the preprocessed text, the corresponding industry vocabulary prediction model is generated and applied to the input system of the industry.

References

[1] Bhuyan M P, Sarma S K. (2019).An N-gram based model for predicting of word-formation in Assamese language [J]. Journal of Information & Optimization Sciences, 40(2):427-440.

[2] Hamarashid H K, Saeed S A, Rashid T A, et al. (2021). Next word prediction based on the N-gram model for Kurdish Sorani and Kurmanji [J]. Neural Computing and Applications, 33(5): 4547-4566.

[3] Le C C, Prasad P, Alsadoon A, et al. (2019). Text Classification: Naieve Bayes Classifier with Sentiment Lexicon[J]. IAENG Internaitonal journal of computer science, 46(2):141-148.

[4] Landauer TK, Foltz PW, Laham D. (1998). An Introduction to Latent Semantic Analysis[J]. Discourse Processes, 25, 259-284.

[5] Atlinar F, Ayar T, Darrige A, et al. (2020). Masked Word Prediction with Statistical and Neural Language Models[C]// 2020 Innovations in Intelligent Systems and Applications Conference (ASYU), 1-4

[6] Hochreiter S, Schmidhuber J. (1997). Long short-term memory[J]. Neural Computation, 9(8):1735-1780.

[7] Y. Liu, Y. Wang, X. Yang and L. Zhang. (2017). Short-term travel time prediction by deep learning: A comparison of different LSTM-DNN models[C]// 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), 1-8.

[8] Nicole M, Klinton B. (2019). Using LSTMs to Assess the Obligatoriness of Phonological Distinctive Features for Phonotactic Learning[C]// Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 1595-1605.

[9] Ang Zifeng, Zeng Zhiqi, Wang Ke, et al. (2020). Modified SEIR and AI prediction of the epidemics trend of COVID-19 in China under public health interventions [J]. Journal of thoracic disease, 12(3): 165.

[10] Yazdani A, Safdari R, Golkar A, et al. (2019). Words prediction based on N-gram model for free-text entry in electronic health records [J]. Health Information Science & Systems, 7(1): 1-7.

[11] Goulart H X, Tosi M, Gonalves D S, et al. (2018). Hybrid Model For Word Prediction Using Naive Bayes and Latent Information [J]. CoRR abs/1803.00985.

[12] Towfique H, Mubin M, Hasan M, et al. (2020). Mutual Context-based Word Prediction for Internet Messenger Chat[C]// In International Conference on Computing Advancements, New York, 1-6.

[13] Mikolov T, Karafiat M, Burget L, et al. (2010). Recurrent neural network based language model[C]// Interspeech, Conference of the International Speech Communication Association, Makuhari, 1045-1048.

[14] Zhou C, SunC, Liu Z, et al. (2015). A C-LSTM Neural Network for Text Classification[J]. Computer Science, 1(4): 39-44.

[15] Wang M, Lu Z, Hang L, et al. (2015). genCNN: A Convolutional Architecture for Word Sequence Prediction[C]// Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, 2015: 1567-1576.

[16] Zhao Lu-cai, Shi Bo, Luo Hai-qiong, et al. (2020). Application of bi-directional LSTM neural network based on Bert feature in Chinese electronic medical record input recommendation [J]. China Digital Medicine, 15(04): 55-57+51.

[17] Sukhbaatar S, Weston J, Fergus R. (2015). End-to-End Memory Networks[C]// In: Advances in Neural Information Processing Systems, Cambridge, 2431-2439.

[18] Ghosh S, Vinyals O, Strope B, et al. (2016). Contextual LSTM (CLSTM) models for Large scale NLP tasks[J]. CoRR abs/1602.06291.

[19] Gre Ff K, Srivastava R K, J Koutnik, et al. (2016). LSTM: A Search Space Odyssey[J]. IEEE Transactions on Neural Networks & Learning Systems, 28(10):2222-2232.

[20] Kingma D P , Ba J L. (2015). Adam: a method for stochastic optimization[C]// 3rd International Conference on Learning Representations, San Diego, 1-25.

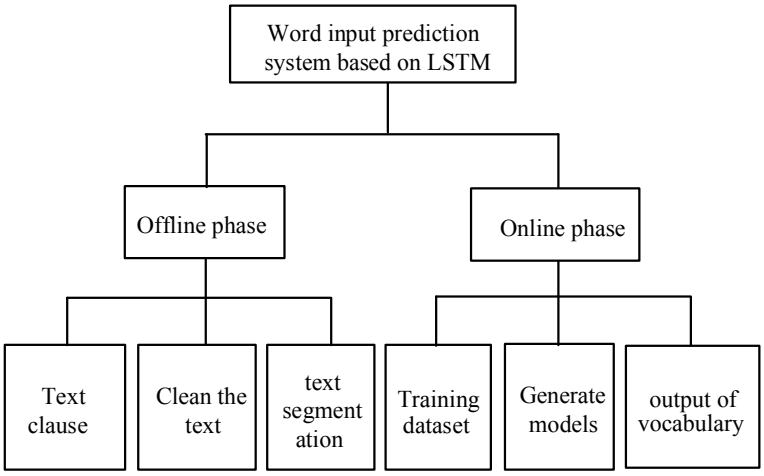

Figure 1. The flow framework of LSTM-based word input prediction system

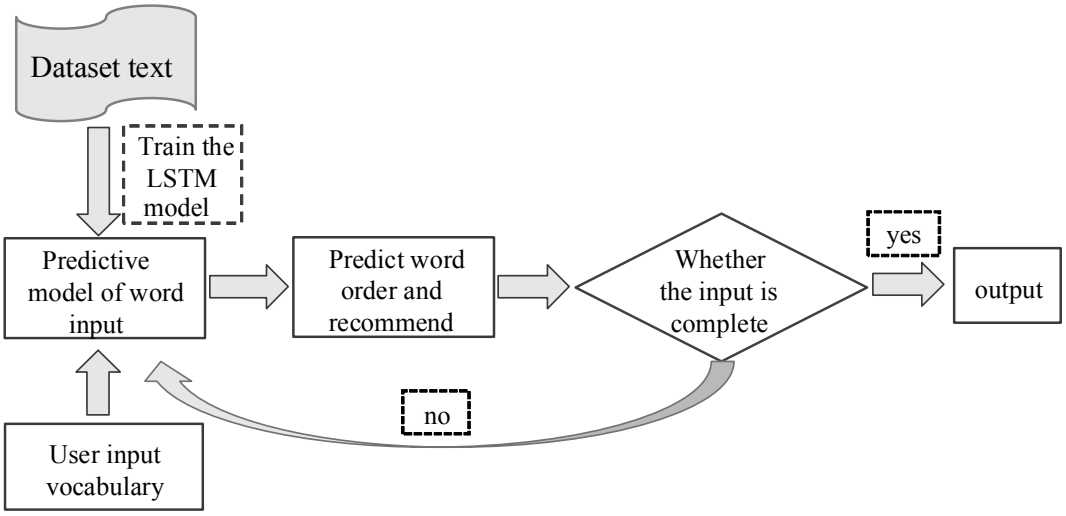

Figure 2. Online phase

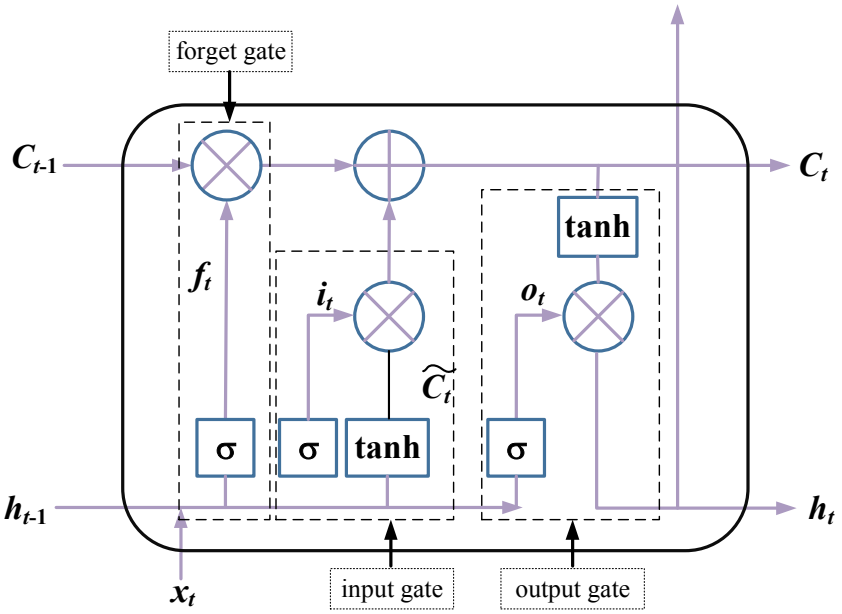

Figure 3. The structure of LSTM

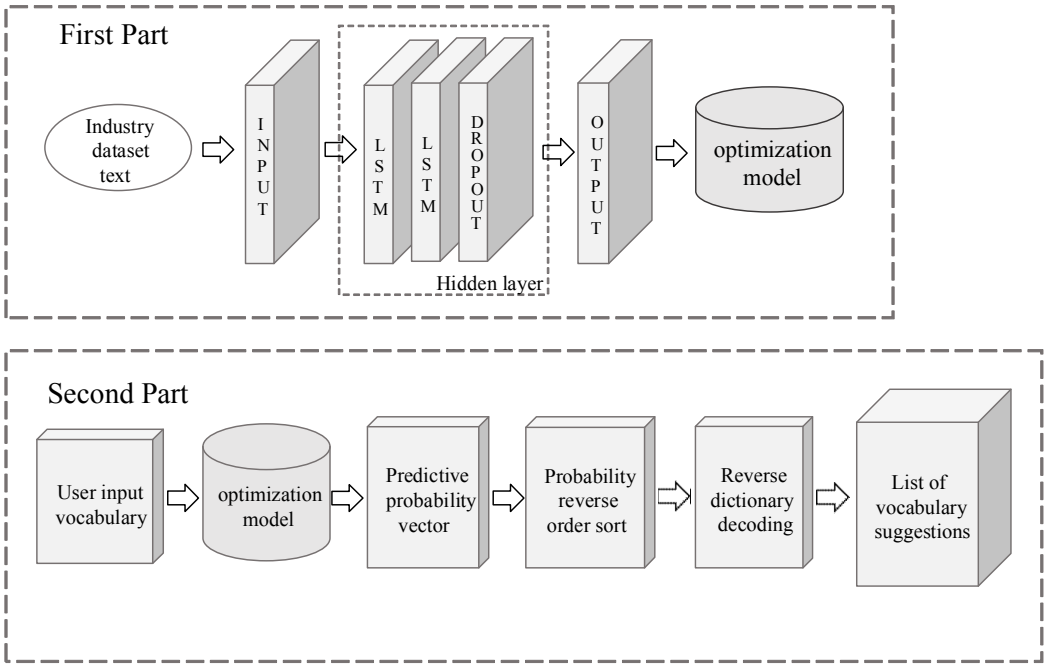

Figure 4.The LSTM-based word input prediction model

Table 1. Word inputs for sports news predict the sequence of words returned

k	world	football	Final
1	record	match	team
2	champion	team	referee
3	cup	player	broadcast
4	class	manager	victory
5	medal	star	midfield

Table 2. Word inputs for financialnews predict the sequence of words returned

k	profit	financial	stock
1	dip	crisis	market
2	rise	market	exchange
3	margin	capital	price
4	sharing	ratios	fund
5	shrinkage	company	certificate